

# 计算机组成习题参考答案

## —流水线处理器设计

❖ 本习题讨论流水线对处理器时钟周期的影响。表中给出了数据通路中不同阶段延迟的两种情况，试根据这两种情况分别回答下列问题。

	IF	ID	EX	MEM	WB
a.	300ps	400ps	350ps	500ps	100ps
b.	200ps	150ps	120ps	190ps	140ps

- 1) 流水线处理器与非流水线处理器的时钟周期分别是什么？
- 2) lw 指令在流水线处理器和非流水线处理器中的总延迟分别是多少？
- 3) 如果可以将原流水线数据通路的一级划分为两级，每级的延迟是原级的一半，会选择哪一级进行划分？划分后处理器的时钟周期为多少？

❖ 本习题讨论流水线对处理器时钟周期的影响。表中给出了数据通路中不同阶段延迟的两种情况，试根据这两种情况分别回答下列问题。

	IF	ID	EX	MEM	WB
a.	300ps	400ps	350ps	500ps	100ps
b.	200ps	150ps	120ps	190ps	140ps

➤ 1) 流水线处理器与非流水线处理器的时钟周期分别是什么？

■ a. 流水线处理器：500ps（MEM），非流水线处理器： $300 + 400 + 350 + 500 + 100 = 1650\text{ps}$

■ b. 流水线处理器：200ps（IF），非流水线处理器： $200 + 150 + 120 + 190 + 140 = 800\text{ps}$

■ 解析：流水线处理器的时钟周期受限于处理速度最慢的一级，非流水线处理器的时钟周期受限于处理时间最长的指令（一般为lw指令）。

❖ 本习题讨论流水线对处理器时钟周期的影响。表中给出了数据通路中不同阶段延迟的两种情况，试根据这两种情况分别回答下列问题。

	IF	ID	EX	MEM	WB
a.	300ps	400ps	350ps	500ps	100ps
b.	200ps	150ps	120ps	190ps	140ps

➤ 2) lw 指令在流水线处理器和非流水线处理器中的总延迟分别是多少？

- a. 流水线处理器：  $500 \times 5 = 2500\text{ps}$ ，非流水线处理器：  $300 + 400 + 350 + 500 + 100 = 1650\text{ps}$
- b. 流水线处理器：  $200 \times 5 = 1000\text{ps}$ ，非流水线处理器：  $200 + 150 + 120 + 190 + 140 = 800\text{ps}$
- 解析：lw指令需要完整执行IF、ID、EX、MEM、WB五级。

❖ 本习题讨论流水线对处理器时钟周期的影响。表中给出了数据通路中不同阶段延迟的两种情况，试根据这两种情况分别回答下列问题。

	IF	ID	EX	MEM	WB
a.	300ps	400ps	350ps	500ps	100ps
b.	200ps	150ps	120ps	190ps	140ps

➤ 3) 如果可以将原流水线数据通路的一级划分为两级，每级的延迟是原级的一半，会选择哪一级进行划分？划分后处理器的时钟周期为多少？

- a. 选择MEM级，划分后得到MEM1和MEM2级的延迟分别为250ps，处理器的时钟周期为400ps（ID）
- b. 选择IF级，划分后得到IF1和IF2级的延迟分别为100ps，处理器的时钟周期为190ps（MEM）
- 解析：应当选择原来处理速度最慢的一级进行划分，得到新的具有六级流水线的处理器，此时处理器的时钟周期仍然受限于处理速度最慢的一级。

❖ 假设处理器执行的指令比例下表两种情况所示，试根据每种情况分别回答下列问题。

	ALU	beq	lw	sw
a.	50%	25%	15%	10%
b.	30%	25%	30%	15%

- 4) 假设没有阻塞和冒险，数据存储器的利用率是多少（占总周期数的百分比）？
- 5) 假设没有阻塞和冒险，寄存器堆的写寄存器端口的利用率是多少？

❖ 假设处理器执行的指令比例下表两种情况所示，试根据每种情况分别回答下列问题。

	ALU	beq	lw	sw
a.	50%	25%	15%	10%
b.	30%	25%	30%	15%

➤ 4) 假设没有阻塞和冒险，数据存储器的利用率是多少（占总周期数的百分比）？

■ a.  $15\% + 10\% = 25\%$

■ b.  $30\% + 15\% = 45\%$

■ 解析：需要用到数据存储器的指令为lw和sw。

❖ 假设处理器执行的指令比例下表两种情况所示，试根据每种情况分别回答下列问题。

	ALU	beq	lw	sw
a.	50%	25%	15%	10%
b.	30%	25%	30%	15%

➤ 5) 假设没有阻塞和冒险，寄存器堆的写寄存器端口的利用率是多少？

■ a.  $50\% + 15\% = 65\%$

■ b.  $30\% + 30\% = 60\%$

■ 解析：需要用到寄存器堆的写寄存器端口的指令为ALU和lw。



❖ 本习题讨论数据相关如何影响基本五级流水线的运行。试根据下表的两种指令序列情况分别回答下列问题。

	指令序列		指令序列
<b>a.</b>	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	<b>b.</b>	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

- 1) 指出指令序列中存在的相关及其类型。
- 2) 假设该流水线处理器没有转发，指出指令序列中存在的冒险并加入nop指令以消除冒险。
- 3) 假设该流水线处理器中有充分的转发。指出指令序列中存在的冒险并加入nop指令以消除冒险。

❖ 本习题讨论数据相关如何影响基本五级流水线的运行。试根据下表的两种指令序列情况分别回答下列问题。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

➤ 1) 指出指令序列中存在的相关及其类型。

- a. I1和I3: RAW \$1, I1和I2: WAR \$6, I2和I3: RAW \$6
- b. I1和I2、I3: RAW \$5, I1、I2和I3: WAR \$5, I1和I3: WAW \$5
- 解析: 三种数据相关类型为: RAW (Read After Write, 写后读), WAR (Write After Read, 读后写) 和WAW (Write After Write, 写后写)。
- a中存在的读写关系: I1写\$1, I3读\$1; I1读\$6, I2写\$6; I2写\$6, I3读\$6。
- b中存在的读写关系: I1写\$5, I2读\$5, I3读\$5; I1读\$5, I2读\$5, I3写\$5; I1写\$5, I3写\$5。

❖ 本习题讨论数据相关如何影响基本五级流水线的运行。试根据下表的两种指令序列情况分别回答下列问题。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

➤ 2) 假设该流水线处理器没有转发，指出指令序列中存在的冒险并加入nop指令以消除冒险。

■ 解析：WAR和WAW不会引起冒险，只有RAW会引起冒险，因而加入nop指令时只需要考虑RAW。lw指令写RF是在WB，sw指令读RF是在ID，add指令读RF是在ID，写RF是在WB，因此至少需要将sw相对于lw推迟两级，将sw相对于add推迟两级，加入nop后的指令序列如下表。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 <b>nop</b> <b>nop</b> sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) <b>nop</b> <b>nop</b> sw \$5, -16(\$5) add \$5, \$5, \$5

❖ 本习题讨论数据相关如何影响基本五级流水线的运行。试根据下表的两种指令序列情况分别回答下列问题。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

➤ 3) 假设该流水线处理器中有充分的转发。指出指令序列中存在的冒险并加入nop指令以消除冒险。

■ 解析：具有了转发之后，ALU可以将计算结果转发至下一条指令的EX，lw可以将MEM结果转发至再下一条指令的EX，这样一来，只需要将sw相对于lw推迟一级即可，而sw通过转发机制可以直接跟在add之后，加入nop后的指令序列如下表。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) <b>nop</b> sw \$5, -16(\$5) add \$5, \$5, \$5

❖ 根据下表的两种时钟周期情况，分别回答下列问题。

	无转发	充分的转发	仅ALU至ALU的转发
a.	300ps	400ps	360ps
b.	200ps	250ps	220ps

- 4) 该指令序列在无转发和充分的转发时总执行时间分别是多少？后者相对于前者的加速比是多少。
- 5) 如果仅有ALU至ALU的转发（没有从MEM到EX的转发），如何加入nop指令以消除可能的冒险？
- 6) 该指令序列在仅有ALU至ALU的转发时总执行时间分别是多少？与无转发的情况相比，加速比是多少？

❖ 根据下表的两种时钟周期情况，分别回答下列问题。

	无转发	充分的转发	仅ALU至ALU的转发
a.	300ps	400ps	360ps
b.	200ps	250ps	220ps

➤ 4) 该指令序列在无转发和充分的转发时总执行时间分别是多少？后者相对于前者的加速比是多少。

■ a. 无转发： $(7 + 2) \times 300 = 2700\text{ps}$ ，充分的转发： $7 \times 400\text{ps} = 2800\text{ps}$ ，加速比： $2700/2800 = 0.96$

■ b. 无转发： $(7 + 2) \times 200 = 1800\text{ps}$ ，充分的转发： $(7 + 1) \times 250 = 2000\text{ps}$ ，加速比： $1800/2000 = 0.90$

■ 解析：无转发时，a中的指令序列需要插入2个nop，b中的指令序列需要插入2个nop（详见2)问）；有充分的转发时，a中的指令序列不需要再插入nop，b中的指令序列需要插入1个nop（详见3)问）。

原始指令序列			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

无转发			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 <b>nop</b> <b>nop</b> sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) <b>nop</b> <b>nop</b> sw \$5, -16(\$5) add \$5, \$5, \$5

充分的转发			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) <b>nop</b> sw \$5, -16(\$5) add \$5, \$5, \$5

❖ 根据下表的两种时钟周期情况，分别回答下列问题。

	无转发	充分的转发	仅ALU至ALU的转发
a.	300ps	400ps	360ps
b.	200ps	250ps	220ps

➤ 5) 如果仅有ALU至ALU的转发（没有从MEM到EX的转发），如何加入nop指令以消除可能的冒险？

■ 解析：若只有ALU至ALU的转发，一条ALU指令可以向下一条指令的EX转发ALU计算结果，但不能向再下一条指令转发，因此实际上此时无法转发lw中MEM的数据，sw相对于lw仍然至少需要推迟两级，但是sw此时通过ALU至ALU的转发机制可以直接跟在add之后，加入nop后的指令序列如下表。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 <b>nop</b> sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) <b>nop</b> <b>nop</b> sw \$5, -16(\$5) add \$5, \$5, \$5

❖ 根据下表的两种时钟周期情况，分别回答下列问题。

	无转发	充分的转发	仅ALU至ALU的转发
a.	300ps	400ps	360ps
b.	200ps	250ps	220ps

➤ 6) 该指令序列在仅有ALU至ALU的转发时总执行时间分别是多少？与无转发的情况相比，加速比是多少？

■ a. 无转发： $(7 + 2) \times 300 = 2700\text{ps}$ ，仅有ALU至ALU的转发： $(7 + 1) \times 360\text{ps} = 2880\text{ps}$ ，加速比： $2700/2880 = 0.94$

■ b. 无转发： $(7 + 2) \times 200 = 1800\text{ps}$ ，仅有ALU至ALU的转发： $(7 + 2) \times 220 = 1980\text{ps}$ ，加速比： $1800/1980 = 0.91$

■ 解析：无转发时，a中的指令序列需要插入2个nop，b中的指令序列需要插入2个nop（详见2)问）；仅有ALU至ALU的转发时，a中的指令序列需要插入1个nop，b中的指令序列需要插入2个nop（详见5)问）。

原始指令序列			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) sw \$5, -16(\$5) add \$5, \$5, \$5

无转发			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 nop nop sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) nop nop sw \$5, -16(\$5) add \$5, \$5, \$5

仅有ALU至ALU的转发			
a.	lw \$1, 40(\$6) add \$6, \$2, \$2 nop sw \$6, 50(\$1)	b.	lw \$5, -16(\$5) nop nop sw \$5, -16(\$5) add \$5, \$5, \$5



❖ 本习题讨论指令集对流水线设计的影响。试根据下表的两条新指令回答下列问题。

a.	bezi (Rs), Label	if Mem[Rs] = 0 then PC = PC + Offs
b.	swi Rd, Rs(Rt)	Mem[Rs+Rt] = Rd

- 1) 为了将这条新指令增加到MIPS指令集，必须对流水线数据通路做什么改动？
- 2) 需要在第1)问的数据通路上增加哪些控制信号？
- 3) 对新指令的支持是否会引入新的冒险？已有冒险导致的阻塞是否会更加严重？

❖ 本习题讨论指令集对流水线设计的影响。试根据下表的两条新指令回答下列问题。

a.	bezi (Rs), Label	if Mem[Rs] = 0 then PC = PC + Offs
b.	swi Rd, Rs(Rt)	Mem[Rs+Rt] = Rd

➤ 1) 为了将这条新指令增加到MIPS指令集，必须对流水线数据通路做什么改动？

- a. 需要在ALU前的MUX增加一个输入端0；需要在WB中增加一个比较数据存储器值与0的比较器（或者专门为比较这个过程增加新的一级流水，but this is not recommended）；需要在PC前的MUX控制信号中加入比较器的结果，并且此时MUX的控制信号必须一直阻塞到WB之后才能进行选择。
- b. 需要为寄存器堆加入一个新的读寄存器地址写入端和读出端；需要在EX中ALU之后加入一个MUX来选择要写入数据存储器的值是Rd（对于swi指令）还是Rt（对于sw指令）。

❖ 本习题讨论指令集对流水线设计的影响。试根据下表的两条新指令回答下列问题。

a.	bezi (Rs), Label	if Mem[Rs] = 0 then PC = PC + Offs
b.	swi Rd, Rs(Rt)	Mem[Rs+Rt] = Rd

➤ 2) 需要在第1)问的数据通路上增加哪些控制信号？

- a. 需要为ALU前的MUX增加1位控制信号位，使其可以选择出新添加的输入端0；需要为PC前的MUX控制信号加入比较器的结果，来决定是否需要修改PC。
- b. 需要为新加入的MUX增加控制信号。

❖ 本习题讨论指令集对流水线设计的影响。试根据下表的两条新指令回答下列问题。

a.	bezi (Rs), Label	if Mem[Rs] = 0 then PC = PC + Offs
b.	swi Rd, Rs(Rt)	Mem[Rs+Rt] = Rd

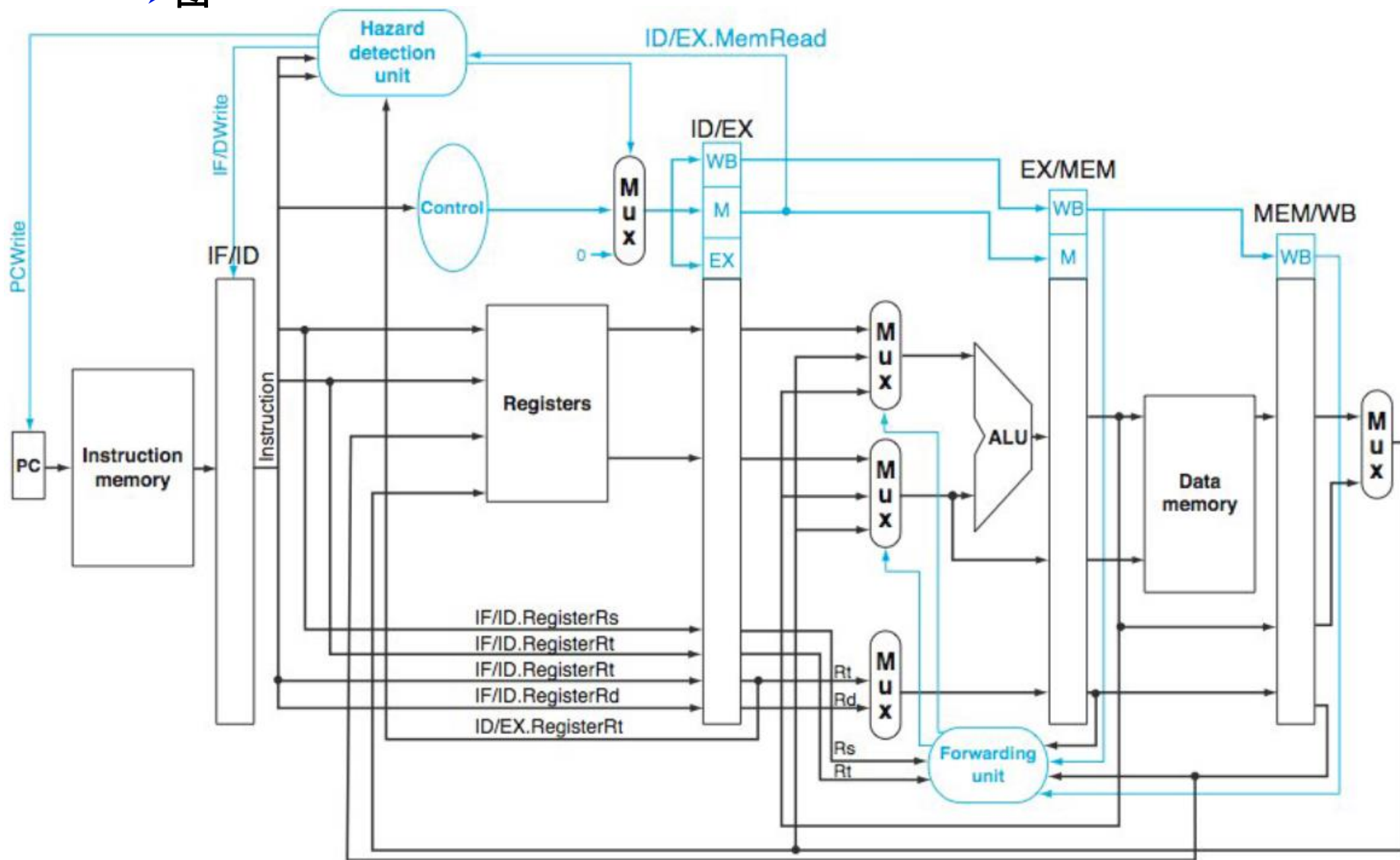
➤ 3) 对新指令的支持是否会引入新的冒险？已有冒险导致的阻塞是否会更加严重？

- a. 会。bezi指令会引入一个新的控制冒险，它的PC直到WB才能确定是否需要修改，而且无法通过缩短分支延迟的方法提前判断出分支是否执行，因而会使已有冒险导致的阻塞更加严重。
- b. 不会。swi指令不改变任何寄存器的值，不会引起数据冒险，同时它也不是分支指令，不会引起控制冒险。

❖ 本习题讨论转发、冒险检测和指令集设计之间的关系。分别根据下表的两个指令序列回答下列问题。假设其在一个五级流水线上执行。

	指令序列		指令序列
<b>a.</b>	lw \$1, 40(\$6) add \$2, \$3, \$1 add \$1, \$6, \$4 sw \$2, 20(\$4) and \$1, \$1, \$4	<b>b.</b>	add \$1, \$5, \$3 sw \$1, 0(\$2) lw \$1, 4(\$2) add \$5, \$5, \$1 sw \$1, 0(\$2)

- 1) 如果没有转发或冒险检测电路，请插入nop指令以保证正确执行。
- 2) 重做第1)问，这次仅当通过改变或重排序指令都不能避免冒险时才插入nop指令。假设可以使用寄存器R7作为临时寄存器。
- 3) 如果处理器中存在转发，但忘了实现冒险检测单元（以为实现了），代码执行时会发生什么情况？
- 4) 如果没有转发，对图中的冒险检测单元来说还需要哪些新的输入输出信号？以该指令序列为例，说明为什么需要这些信号。



❖ 本习题讨论转发、冒险检测和指令集设计之间的关系。分别根据下表的两个指令序列回答下列问题。假设其在一个五级流水线上执行。

	指令序列		指令序列
a.	lw \$1, 40(\$6) add \$2, \$3, \$1 add \$1, \$6, \$4 sw \$2, 20(\$4) and \$1, \$1, \$4	b.	add \$1, \$5, \$3 sw \$1, 0(\$2) lw \$1, 4(\$2) add \$5, \$5, \$1 sw \$1, 0(\$2)

➤ 1) 如果没有转发或冒险检测电路，请插入nop指令以保证正确执行。

	指令序列		指令序列
a.	lw <b>\$1</b> , 40(\$6) <b>nop</b> <b>nop</b> add <b>\$2</b> , \$3, <b>\$1</b> add \$1, \$6, \$4 <b>nop</b> sw <b>\$2</b> , 20(\$4) and \$1, \$1, \$4	b.	add <b>\$1</b> , \$5, \$3 <b>nop</b> <b>nop</b> sw <b>\$1</b> , 0(\$2) lw <b>\$1</b> , 4(\$2) <b>nop</b> <b>nop</b> add \$5, \$5, <b>\$1</b> sw \$1, 0(\$2)

❖ 本习题讨论转发、冒险检测和指令集设计之间的关系。分别根据下表的两个指令序列回答下列问题。假设其在一个五级流水线上执行。

	指令序列		指令序列
<b>a.</b>	I1: lw \$1, 40(\$6) I2: add \$2, \$3, \$1 I3: add \$1, \$6, \$4 I4: sw \$2, 20(\$4) I5: and \$1, \$1, \$4	<b>b.</b>	I1: add \$1, \$5, \$3 I2: sw \$1, 0(\$2) I3: lw \$1, 4(\$2) I4: add \$5, \$5, \$1 I5: sw \$1, 0(\$2)

➤ 2) 重做第1)问，这次仅当通过改变或重排序指令都不能避免冒险时才插入nop指令。假设可以使用寄存器R7作为临时寄存器。

	指令序列		指令序列
<b>a.</b>	I1: lw \$7, 40(\$6) I3: add \$1, \$6, \$4 <b>nop</b> I2: add \$2, \$3, \$7 I5: and \$1, \$1, \$4 <b>nop</b> I4: sw \$2, 20(\$4)	<b>b.</b>	I1: add \$7, \$5, \$3 I3: lw \$1, 4(\$2) <b>nop</b> I2: sw \$7, 0(\$2) I4: add \$5, \$5, \$1 I5: sw \$1, 0(\$2)



❖ 本习题讨论转发、冒险检测和指令集设计之间的关系。分别根据下表的两个指令序列回答下列问题。假设其在一个五级流水线上执行。

	指令序列		指令序列
<b>a.</b>	I1: lw \$1, 40(\$6) I2: add \$2, \$3, \$1 I3: add \$1, \$6, \$4 I4: sw \$2, 20(\$4) I5: and \$1, \$1, \$4	<b>b.</b>	I1: add \$1, \$5, \$3 I2: sw \$1, 0(\$2) I3: lw \$1, 4(\$2) I4: add \$5, \$5, \$1 I5: sw \$1, 0(\$2)

➤ 3) 如果处理器中存在转发，但忘了实现冒险检测单元（以为实现了），代码执行时会发生什么情况？

- a. I2获得的\$1的值是I1前面的那一条指令中\$1的值，此时\$1的值还没有被lw指令更新。
- b. I4获得的\$1的值是I3前面的那一条指令（即I2）中\$1的值，此时\$1的值还没有被lw指令更新。

转发单元可以解决EX冒险和MEM冒险，但是无法解决load-use型数据冒险；冒险检测单元用来检测load-use型数据冒险，在load指令与紧随其后利用他的结果的指令间插入阻塞。

❖ 本习题讨论转发、冒险检测和指令集设计之间的关系。分别根据下表的两个指令序列回答下列问题。假设其在一个五级流水线上执行。

	指令序列		指令序列
<b>a.</b>	lw \$1, 40(\$6) add \$2, \$3, \$1 add \$1, \$6, \$4 sw \$2, 20(\$4) and \$1, \$1, \$4	<b>b.</b>	add \$1, \$5, \$3 sw \$1, 0(\$2) lw \$1, 4(\$2) add \$5, \$5, \$1 sw \$1, 0(\$2)

➤ 4) 如果没有转发，对图中的冒险检测单元来说还需要哪些新的输入输出信号？以该指令序列为例，说明为什么需要这些信号。

- 输入信号：冒险检测单元需要在EX中检查R型指令和lw指令的Rd寄存器，在MEM中检查目标寄存器号，因此需要添加ID/EX流水线寄存器的Rd和EX/MEM的输出寄存器作为输入信号，图4-60中已经有ID/EX的Rt这个输入信号了，不需要再次添加。
- 输出信号：不需要额外的输出信号。